

日本国特許庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出願年月日

Date of Application:

2002年 8月30日

出願番号

Application Number:

特願2002-255258

[ST.10/C]:

[JP2002-255258]

出願人

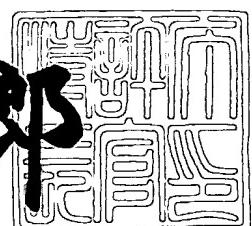
Applicant(s):

セイコーエプソン株式会社

2003年 6月16日

特許庁長官  
Commissioner,  
Japan Patent Office

太田 信一郎



出証番号 出証特2003-3046994

【書類名】 特許願  
【整理番号】 J0092594  
【あて先】 特許庁長官殿  
【国際特許分類】 G09G 5/24  
G09G 5/26

【発明者】

【住所又は居所】 長野県諏訪市大和3丁目3番5号 セイコーエプソン株式会社内  
【氏名】 胡桃澤 孝

【発明者】

【住所又は居所】 長野県諏訪市大和3丁目3番5号 セイコーエプソン株式会社内  
【氏名】 桃園 幸信

【特許出願人】

【識別番号】 000002369  
【氏名又は名称】 セイコーエプソン株式会社

【代理人】

【識別番号】 100095728

【弁理士】

【氏名又は名称】 上柳 雅誉

【連絡先】 0266-52-3139

【選任した代理人】

【識別番号】 100107076

【弁理士】

【氏名又は名称】 藤綱 英吉

【選任した代理人】

【識別番号】 100107261

【弁理士】

【氏名又は名称】 須澤 修

【手数料の表示】

【予納台帳番号】 013044

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 0109826

【ブルーフの要否】 要

【書類名】 明細書

【発明の名称】 フォント処理装置、端末装置、フォント処理方法およびフォント処理プログラム

【特許請求の範囲】

【請求項1】 フォント変形指示を受け取る手段と、  
ビットマップフォントのフォントデータを取得するデータ取得手段と、  
前記フォント変形指示がフォントの横方向の変形を含む場合、前記フォントデータの画素構成を縦方向に分析し、分析結果に基づいて前記フォントデータを横方向に拡大又は縮小する横方向変形手段と、

前記フォント変形指示がフォントの縦方向の変形を含む場合、前記フォントデータの画素構成を横方向に分析し、分析結果に基づいて前記フォントデータを縦方向に拡大又は縮小する縦方向変形手段と、を備えることを特徴とするフォント処理装置。

【請求項2】 前記横方向変形手段は、  
前記フォントデータを複数の行に分割する分割手段と、  
前記行ごとに、画素構成の特徴を示すコストを計算する計算手段と、  
前記コストの小さい行から順に、所定数の行について画素の拡大又は縮小処理を行う手段と、を備えることを特徴とする請求項1に記載のフォント処理装置。

【請求項3】 前記横方向変形手段は、  
前記フォントデータを複数の行に分割する分割手段と、  
前記行ごとに、画素構成の特徴を示すコストを計算する計算手段と、  
前記コストの大きい行から順に、所定数の行について画素の拡大又は縮小処理を行う手段と、を備えることを特徴とする請求項1に記載のフォント処理装置。

【請求項4】 前記縦方向変形手段は、  
前記フォントデータを複数の列に分割する分割手段と、  
前記列ごとに、画素構成の特徴を示すコストを計算する計算手段と、  
前記コストの小さい列から順に、所定数の列について画素の拡大又は縮小処理を行う手段と、を備えることを特徴とする請求項1乃至3のいずれか一項に記載のフォント処理装置。

【請求項5】 前記縦方向変形手段は、

前記フォントデータを複数の列に分割する分割手段と、

前記列ごとに、画素構成の特徴を示すコストを計算する計算手段と、

前記コストの大きい列から順に、所定数の列について画素の拡大又は縮小処理を行う手段と、を備えることを特徴とする請求項1乃至3のいずれか一項に記載のフォント処理装置。

【請求項6】 縦方向及び横方向の変形率を含むフォント変形指示を受け取る手段と、

ピットマップフォントのフォントデータを取得するデータ取得手段と、

前記横方向の変形率に応じて、前記フォントデータの横方向の変形処理を行う横方向変形手段と、

前記縦方向の変形率に応じて、前記フォントデータの縦方向の変形処理を行う縦方向変形手段と、を備え、

前記横方向変形手段は、

前記フォントデータを複数の行に分割する手段と、

前記行ごとに、画素構成の特徴を示すコストを計算する第1のコスト計算手段と、

前記コストの小さい行から順に、第1の所定数の行を横方向に拡大又は縮小する手段と、を備え、

前記縦方向変形手段は、

前記フォントデータを複数の列に分割する手段と、

前記列ごとに、画素構成の特徴を示すコストを計算する第2のコスト計算手段と、

前記コストの小さい列から順に、第2の所定数の列を縦方向に拡大又は縮小する手段と、を備えることを特徴とするフォント処理装置。

【請求項7】 縦方向及び横方向の変形率を含むフォント変形指示を受け取る手段と、

ピットマップフォントのフォントデータを取得するデータ取得手段と、

前記横方向の変形率に応じて、前記フォントデータの横方向の変形処理を行う

横方向変形手段と、

前記縦方向の変形率に応じて、前記フォントデータの縦方向の変形処理を行う  
縦方向変形手段と、を備え、

前記横方向変形手段は、

前記フォントデータを複数の行に分割する手段と、

前記行ごとに、画素構成の特徴を示すコストを計算する第1のコスト計算手段  
と、

前記コストの大きい行から順に、第1の所定数の行を横方向に拡大又は縮小す  
る手段と、を備え、

前記縦方向変形手段は、

前記フォントデータを複数の列に分割する手段と、

前記列ごとに、画素構成の特徴を示すコストを計算する第2のコスト計算手段  
と、

前記コストの大きい列から順に、第2の所定数の列を縦方向に拡大又は縮小す  
る手段と、を備えることを特徴とするフォント処理装置。

【請求項8】 前記第1の所定数は、前記フォントデータの縦方向及び横方  
向の画素数と前記横方向の変形率とにより決定され、

前記第2の所定数は、前記フォントデータの縦方向及び横方向の画素数と前記  
縦方向の変形率とにより決定されることを特徴とする請求項6又は7に記載のフ  
ォント処理装置。

【請求項9】 前記コスト計算手段は、前記変形処理が拡大である場合には  
、前記行又は列ごとに前記画素構成の線分量を示すコストを計算し、前記変形処  
理が縮小である場合には、前記行又は列ごとに当該行又は列の画素構成と隣接す  
る行又は列の画素構成との類似度を示すコストを計算することを特徴とする請求  
項6乃至8のいずれか一項に記載のフォント処理装置。

【請求項10】 前記コスト計算手段は、前記線分量を示すコストを計算す  
る場合には、前記行又は列ごとに、当該行又は列を構成する画素数をコストとし  
て計算し、さらに、当該行又は列の方向に隣接画素を有する画素数を前記コスト  
に加算することを特徴とする請求項9に記載のフォント処理装置。

【請求項11】 前記コスト計算手段は、前記線分量を示すコストを計算する場合には、前記行又は列ごとに、当該行又は列を構成する画素数を計算し、さらに、当該行又は列の方向に隣接画素を有する画素数を前記当該行又は列を構成する画素数に加算し、加算された結果の逆数をコストとして計算することを特徴とする請求項9に記載のフォント処理装置。

【請求項12】 前記コスト計算手段は、前記類似度を示すコストを計算する場合には、前記行又は列ごとに、当該行又は列の画素構成と、当該行又は列に隣接する行又は列の画素構成との排他的論理和に基づいてコストを計算することを特徴とする請求項9に記載のフォント処理装置。

【請求項13】 前記フォントデータは、当該フォントデータの変形処理において縦方向の変形処理と横方向の変形処理を実行すべき順序を示す処理順序情報を含み、

前記処理順序情報に従って、前記横方向の変形処理と前記縦方向の変形処理の実行順序を制御する順序制御手段をさらに備えることを特徴とする請求項6乃至12のいずれか一項に記載のフォント処理装置。

【請求項14】 前記請求項1乃至13のいずれか一項のフォント処理装置と、

前記フォント処理装置により生成されたフォントデータを記憶する記憶手段と、

前記フォント処理装置により生成されたフォントデータを表示する表示部と、を備えることを特徴とする端末装置。

【請求項15】 フォント変形指示を受け取る工程と、

ビットマップフォントのフォントデータを取得するデータ取得工程と、

前記フォント変形指示がフォントの横方向の変形を含む場合、前記フォントデータの画素構成を縦方向に分析し、分析結果に基づいて前記フォントデータを横方向に拡大又は縮小する横方向変形工程と、

前記フォント変形指示がフォントの横方向の変形を含む場合、前記フォントデータの画素構成を横方向に分析し、分析結果に基づいて前記フォントデータを縦方向に拡大又は縮小する縦方向変形工程と、を有することを特徴とするフォント

処理方法。

【請求項16】 縦方向及び横方向の変形率を含むフォント変形指示を受け取る工程と、

ビットマップフォントのフォントデータを取得するデータ取得工程と、

前記横方向の変形率に応じて、前記フォントデータの横方向の変形処理を行う横方向変形工程と、

前記縦方向の変形率に応じて、前記フォントデータの縦方向の変形処理を行う縦方向変形工程と、を有し、

前記横方向変形工程は、

前記フォントデータを複数の行に分割する工程と、

前記行ごとに、画素構成の特徴を示すコストを計算する第1のコスト計算工程と、

前記コストの小さい行から順に、第1の所定数の行を横方向に拡大又は縮小する工程と、を有し、

前記縦方向変形工程は、

前記フォントデータを複数の列に分割する工程と、

前記列ごとに、画素構成の特徴を示すコストを計算する第2のコスト計算工程と、

前記コストの小さい列から順に、第2の所定数の列を縦方向に拡大又は縮小する工程と、を有することを特徴とするフォント処理方法。

【請求項17】 縦方向及び横方向の変形率を含むフォント変形指示を受け取る工程と、

ビットマップフォントのフォントデータを取得するデータ取得工程と、

前記横方向の変形率に応じて、前記フォントデータの横方向の変形処理を行う横方向変形工程と、

前記縦方向の変形率に応じて、前記フォントデータの縦方向の変形処理を行う縦方向変形工程と、を有し、

前記横方向変形工程は、

前記フォントデータを複数の行に分割する工程と、

前記行ごとに、画素構成の特徴を示すコストを計算する第1のコスト計算工程と、

前記コストの大きい行から順に、第1の所定数の行を横方向に拡大又は縮小する工程と、を有し、

前記縦方向変形工程は、

前記フォントデータを複数の列に分割する工程と、

前記列ごとに、画素構成の特徴を示すコストを計算する第2のコスト計算工程と、

前記コストの大きい列から順に、第2の所定数の列を縦方向に拡大又は縮小する工程と、を有することを特徴とするフォント処理方法。

【請求項18】 コンピュータを有する端末装置により実行されるフォント処理プログラムであって、前記コンピュータを、

フォント変形指示を受け取る手段、

ビットマップフォントのフォントデータを取得するデータ取得手段、

前記フォント変形指示がフォントの横方向の変形を含む場合、前記フォントデータの画素構成を縦方向に分析し、分析結果に基づいて前記フォントデータを横方向に拡大又は縮小する横方向変形手段、

前記フォント変形指示がフォントの横方向の変形を含む場合、前記フォントデータの画素構成を横方向に分析し、分析結果に基づいて前記フォントデータを縦方向に拡大又は縮小する縦方向変形手段、として機能させることを特徴とするフォント処理プログラム。

【請求項19】 コンピュータを有する端末装置により実行されるフォント処理プログラムであって、前記コンピュータを、

縦方向及び横方向の変形率を含むフォント変形指示を受け取る手段、

ビットマップフォントのフォントデータを取得するデータ取得手段、

前記横方向の変形率に応じて、前記フォントデータの横方向の変形処理を行う横方向変形手段、

前記縦方向の変形率に応じて、前記フォントデータの縦方向の変形処理を行う縦方向変形手段として機能させ、

前記横方向変形手段は、

前記フォントデータを複数の行に分割する手段と、

前記行ごとに、画素構成の特徴を示すコストを計算する第1のコスト計算手段と、

前記コストの小さい行から順に、第1の所定数の行を横方向に拡大又は縮小する手段と、を備え、

前記縦方向変形手段は、

前記フォントデータを複数の列に分割する手段と、

前記列ごとに、画素構成の特徴を示すコストを計算する第2のコスト計算手段と、

前記コストの小さい列から順に、第2の所定数の列を縦方向に拡大又は縮小する手段と、を備えることを特徴とするフォント処理プログラム。

【請求項20】 コンピュータを有する端末装置により実行されるフォント処理プログラムであって、前記コンピュータを、

縦方向及び横方向の変形率を含むフォント変形指示を受け取る手段、

ビットマップフォントのフォントデータを取得するデータ取得手段、

前記横方向の変形率に応じて、前記フォントデータの横方向の変形処理を行う横方向変形手段、

前記縦方向の変形率に応じて、前記フォントデータの縦方向の変形処理を行う縦方向変形手段として機能させ、

前記横方向変形手段は、

前記フォントデータを複数の行に分割する手段と、

前記行ごとに、画素構成の特徴を示すコストを計算する第1のコスト計算手段と、

前記コストの大きい行から順に、第1の所定数の行を横方向に拡大又は縮小する手段と、を備え、

前記縦方向変形手段は、

前記フォントデータを複数の列に分割する手段と、

前記列ごとに、画素構成の特徴を示すコストを計算する第2のコスト計算手段

と、

前記コストの大きい列から順に、第2の所定数の列を縦方向に拡大又は縮小する手段と、を備えることを特徴とするフォント処理プログラム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、ビットマップフォントの変形処理に関する。

【0002】

【背景技術】

携帯電話やPDA (Personal Digital Assistant)などの装置では文字などの表示にビットマップフォントが使用される。ビットマップフォントは、予め用意された画素の配列パターンにより文字や記号などを表示するものである。ベクトルデータの集合として文字や記号などを表示するアウトラインフォントと異なり、ビットマップフォントは単純な画素の配列パターンであるため、1文字当たりのデータ量が小さい。そのため、表示エリアの画素数が比較的少ない携帯電話やPDAなどにおいては、ビットマップフォントが使用される。

【0003】

【発明が解決しようとする課題】

表示装置上に文字や記号を表示する場合、文字の変形が要求されることがある。例えば、標準的な縦横比が等しいフォントを、縦長、横長に変形して表示させる要求が生じうる。このような要求は、例えば縦長や横長の文字が見やすいなどというユーザの好みによる場合もある。また、表示装置が有する表示エリアの形状に依存し、又は、表示すべき画像の構成に依存し、主として表示装置の処理上、そのような要求が生じる場合がある。例えば表示エリアが縦長であるので文字も縦長に表示した方が見やすいとか、表示エリア内に文章と一緒に画像を表示するため、画像の表示面積や表示位置との関係で文字の形状を変形させて表示したいなどの要求が生じうる。

【0004】

携帯電話やPDAなどに使用されるビットマップフォントの場合、1文字のサ

イズが小さく、文字の太さが1画素で表されることも多い。そのような場合、フォントを変形して表示するために、元の文字サイズと拡大後の文字サイズとの縦横比に基づいて単純な座標変換を行う手法によりビットマップフォントを拡大すると、拡大後の文字に不自然な太さの部分ができてしまうことがある。また、同様の小さなビットマップフォントを上記のような単純な座標変換により縮小すると、縮小後に文字が潰れ、不自然な太さの線ができたり、文字の部分同士が不適切に連結して違う文字に見えてしまうことがある。

#### 【0005】

本発明は、以上の点に鑑みてなされたものであり、携帯電話やPDAなどで使用するサイズの小さなビットマップフォントを自然に変形することを課題とする。

#### 【0006】

##### 【課題を解決するための手段】

本発明の1つの観点では、フォント処理装置は、フォント変形指示を受け取る手段と、ビットマップフォントのフォントデータを取得するデータ取得手段と、前記フォント変形指示がフォントの横方向の変形を含む場合、前記フォントデータの画素構成を縦方向に分析し、分析結果に基づいて前記フォントデータを横方向に拡大又は縮小する横方向変形手段と、前記フォント変形指示がフォントの縦方向の変形を含む場合、前記フォントデータの画素構成を横方向に分析し、分析結果に基づいて前記フォントデータを縦方向に拡大又は縮小する縦方向変形手段と、を備える。

#### 【0007】

また、同様の観点によるフォント処理方法は、フォント変形指示を受け取る工程と、ビットマップフォントのフォントデータを取得するデータ取得工程と、前記フォント変形指示がフォントの横方向の変形を含む場合、前記フォントデータの画素構成を縦方向に分析し、分析結果に基づいて前記フォントデータを横方向に拡大又は縮小する横方向変形工程と、前記フォント変形指示がフォントの縦方向の変形を含む場合、前記フォントデータの画素構成を横方向に分析し、分析結果に基づいて前記フォントデータを縦方向に拡大又は縮小する縦方向変形工程と

、を有する。

#### 【0008】

上記のフォント処理装置又は方法は、ビットマップフォントを所定の拡大率又は縮小率で拡大又は縮小することによりフォントを変形する。まず、フォント変形指示を受け取ると、変形の対象となるビットマップフォントのフォントデータを取得する。フォント変形指示が横方向の変形を含む場合、フォントデータを縦方向に分析し、その分析結果に基づいて、フォントデータを横方向に拡大又は縮小する。同様に、フォント変形指示が縦方向の変形を含む場合、フォントデータを横方向に分析し、その分析結果に基づいてフォントデータを縦方向に拡大又は縮小する。これにより、変形の対象となるフォントデータの画素構成に応じて、それぞれ縦方向及び横方向に適切な拡大／縮小処理がなされてフォントが変形されるので、変形処理後のフォントデータが不自然となることを防止できる。

#### 【0009】

上記のフォント処理装置の一態様では、前記横方向変形手段は、前記フォントデータを複数の行に分割する分割手段と、前記行ごとに、画素構成の特徴を示すコストを計算する計算手段と、前記コストの小さい行又は大きい行から順に、所定数の行について画素の拡大又は縮小処理を行う手段と、を備える。

#### 【0010】

この態様によれば、フォントデータを構成する画素の行方向（縦方向）の画素構成が数値化されてコストとして算出され、その値に基づいて拡大／縮小処理が行われる。よって、画素構成の分析を単純なコスト計算により行うことができ、迅速な処理が可能となる。

#### 【0011】

上記のフォント処理装置の他の一態様では、前記縦方向変形手段は、前記フォントデータを複数の列に分割する分割手段と、前記列ごとに、画素構成の特徴を示すコストを計算する計算手段と、前記コストの小さい列又は大きい列から順に、所定数の列について画素の拡大又は縮小処理を行う手段と、を備える。

#### 【0012】

この態様では、フォントデータを構成する画素の列方向（横方向）の画素構成

が数値化されてコストとして算出され、その値に基づいて拡大／縮小処理が行われる。よって、画素構成の分析を単純なコスト計算により行うことができ、迅速な処理が可能となる。

#### 【0013】

本発明の他の観点では、フォント処理装置は、縦方向及び横方向の変形率を含むフォント変形指示を受け取る手段と、ビットマップフォントのフォントデータを取得するデータ取得手段と、前記横方向の変形率に応じて、前記フォントデータの横方向の変形処理を行う横方向変形手段と、前記縦方向の変形率に応じて、前記フォントデータの縦方向の変形処理を行う縦方向変形手段と、を備え、前記横方向変形手段は、前記フォントデータを複数の行に分割する手段と、前記行ごとに、画素構成の特徴を示すコストを計算する第1のコスト計算手段と、前記コストの小さい行又は大きい行から順に、第1の所定数の行を横方向に拡大又は縮小する手段と、を備え、前記縦方向変形手段は、前記フォントデータを複数の列に分割する手段と、前記列ごとに、画素構成の特徴を示すコストを計算する第2のコスト計算手段と、前記コストの小さい列又は大きい列から順に、第2の所定数の列を縦方向に拡大又は縮小する手段と、を備える。

#### 【0014】

また、同様の観点によるフォント処理方法は、縦方向及び横方向の変形率を含むフォント変形指示を受け取る工程と、ビットマップフォントのフォントデータを取得するデータ取得工程と、前記横方向の変形率に応じて、前記フォントデータの横方向の変形処理を行う横方向変形工程と、前記縦方向の変形率に応じて、前記フォントデータの縦方向の変形処理を行う縦方向変形工程と、を有し、前記横方向変形工程は、前記フォントデータを複数の行に分割する工程と、前記行ごとに、画素構成の特徴を示すコストを計算する第1のコスト計算工程と、前記コストの小さい行又は大きい行から順に、第1の所定数の行を横方向に拡大又は縮小する工程と、を有し、前記縦方向変形工程は、前記フォントデータを複数の列に分割する工程と、前記列ごとに、画素構成の特徴を示すコストを計算する第2のコスト計算工程と、前記コストの小さい列又は大きい列から順に、第2の所定数の列を縦方向に拡大又は縮小する工程と、を有する。

## 【0015】

上記のフォント処理装置又は方法によれば、フォントデータが複数の行に分割され、その行ごとにコストが計算される。また、同様に、フォントデータが複数の列に分割され、その列ごとにコストが計算される。そして、コストに応じて、所定数の行及び列を拡大又は縮小することによりフォントの変形がなされる。

## 【0016】

上記のフォント処理装置の一態様では、前記第1の所定数は、前記フォントデータの縦方向及び横方向の画素数と前記横方向の変形率とにより決定され、前記第2の所定数は、前記フォントデータの縦方向及び横方向の画素数と前記縦方向の変形率とにより決定される。よって、フォント変形指示により指定された変形率となるように、必要数の行及び列が拡大／縮小される。

## 【0017】

上記のフォント処理装置のさらに他の一態様では、前記コスト計算手段は、前記変形処理が拡大である場合には、前記行又は列ごとに前記画素構成の線分量を示すコストを計算し、前記変形処理が縮小である場合には、前記行又は列ごとに当該行又は列の画素構成と隣接する行又は列の画素構成との類似度を示すコストを計算する。

## 【0018】

この態様によれば、フォントの変形のために拡大を行う場合、コストは画素の線分量、即ち、その行を構成する画素が単なる点又は点の集合に近いか、線分に近いかを示す指標となる。そして、コストの小さい順に行又は列を拡大する。よって、点や点の集合に近い画素構成を有する列が優先的に拡大され、線分に近い画素構成を有する列は拡大されにくくなる。また、フォントの変形のために縮小を行う場合、コストは隣接する行の画素構成との類似度を示す。そして、コストの小さい順に行を横方向に拡大する。これにより、画素構成が類似する行が隣接する部分が優先的に縮小されるので、文字全体のバランスがくずれなくなる。こうして、自然なフォント変形が実現される。

## 【0019】

上記のフォント処理装置のさらに他の一態様では、前記コスト計算手段は、前

記線分量を示すコストを計算する場合には、前記行又は列ごとに、当該行又は列を構成する画素数をコストとして計算し、さらに、当該行又は列の方向に隣接画素を有する画素数を前記コストに加算する。

## 【0020】

また、上記のフォント処理装置のさらに他の一態様では、前記コスト計算手段は、前記線分量を示すコストを計算する場合には、前記行又は列ごとに、当該行又は列を構成する画素数を計算し、さらに、当該行又は列の方向に隣接画素を有する画素数を前記当該行又は列を構成する画素数に加算し、加算された結果の逆数をコストとして計算する。

## 【0021】

また、上記のフォント処理装置のさらに他の一態様では、前記コスト計算手段は、前記類似度を示すコストを計算する場合には、前記行又は列ごとに、当該行又は列の画素構成と、当該行又は列に隣接する行又は列の画素構成との排他的論理和に基づいてコストを計算する。

## 【0022】

これらの態様によれば、コスト計算手段は、単純な加算処理によってコストを計算するので、フォント拡大処理を簡素な構成で迅速に行うことができる。

## 【0023】

上記のフォント処理装置のさらに他の一態様では、前記フォントデータは、当該フォントデータの変形処理において縦方向の変形処理と横方向の変形処理を実行すべき順序を示す処理順序情報を含み、前記処理順序情報に従って、前記横方向の変形処理と前記縦方向の変形処理の実行順序を制御する順序制御手段をさらに備える。

## 【0024】

この態様によれば、変形の対象となるフォントごとに、縦方向の変形処理と横方向の変形処理のどちらを先に行った方がより自然な変形フォントが得られるかが予め決定され、その順序を示す処理順序情報がフォントデータに付属している。よって、フォント処理装置は、その処理順序情報を参照して縦方向の変形処理と横方向の変形処理の実行順序を決定することにより、自然な変形フォントが得

られる。

## 【0025】

本発明のさらに他の観点では、上記のフォント処理装置と、前記フォント処理装置により生成されたフォントデータを記憶する記憶手段と、前記フォント処理装置により生成されたフォントデータを表示する表示部と、を備える端末装置を構成することができる。当該端末装置は、例えば携帯電話やPDAなどの携帯型端末装置とすることができる。本発明のフォント処理装置により、単純な演算で迅速にフォントの変形ができるので、そのような携帯型端末装置においても自然なフォント変形を行うことができる。

## 【0026】

本発明のさらに他の観点では、コンピュータを有する端末装置により実行されるフォント処理プログラムは、前記コンピュータを、フォント変形指示を受け取る手段、ビットマップフォントのフォントデータを取得するデータ取得手段、前記フォント変形指示がフォントの横方向の変形を含む場合、前記フォントデータの画素構成を縦方向に分析し、分析結果に基づいて前記フォントデータを横方向に拡大又は縮小する横方向変形手段、前記フォント変形指示がフォントの横方向の変形を含む場合、前記フォントデータの画素構成を横方向に分析し、分析結果に基づいて前記フォントデータを縦方向に拡大又は縮小する縦方向変形手段、として機能させる。

## 【0027】

また、本発明のさらに他の観点では、コンピュータを有する端末装置により実行されるフォント処理プログラムは、前記コンピュータを、縦方向及び横方向の変形率を含むフォント変形指示を受け取る手段、ビットマップフォントのフォントデータを取得するデータ取得手段、前記横方向の変形率に応じて、前記フォントデータの横方向の変形処理を行う横方向変形手段、前記縦方向の変形率に応じて、前記フォントデータの縦方向の変形処理を行う縦方向変形手段として機能させ、前記横方向変形手段は、前記フォントデータを複数の行に分割する手段と、前記行ごとに、画素構成の特徴を示すコストを計算する第1のコスト計算手段と、前記コストの小さい行又は大きい行から順に、第1の所定数の行を横方向に拡

大又は縮小する手段と、を備え、前記縦方向変形手段は、前記フォントデータを複数の列に分割する手段と、前記列ごとに、画素構成の特徴を示すコストを計算する第2のコスト計算手段と、前記コストの小さい列又は大きい列から順に、第2の所定数の列を縦方向に拡大又は縮小する手段と、を備える。

## 【0028】

上記のフォント処理プログラムを、端末装置内のコンピュータで実行することにより、上記のフォント処理装置を実現することができる。

## 【0029】

## 【発明の実施の形態】

以下、図面を参照して本発明の好適な実施の形態について説明する。

## 【0030】

## 【携帯端末装置の構成】

図1に、本発明の実施形態にかかるビットマップフォントの変形処理を適用した携帯端末装置の概略構成を示す。図1において、携帯端末装置10は、例えば携帯電話やPDAなど、画像表示エリアが比較的小さい端末装置である。携帯端末装置10は、表示部12と、変形フォントメモリ14と、CPU16と、入力部18と、プログラムROM20と、フォントROM22と、RAM24とを備える。

## 【0031】

表示部22は、例えばLCD (Liquid Crystal Display: 液晶表示装置)などの軽量、薄型の表示装置とすることができ、表示エリア内にビットマップフォントにより構成される文字を表示する。

## 【0032】

入力部18は、携帯電話であれば各種の操作ボタンなど、PDAであればタッチペンなどによる接触を検出するタブレットなどにより構成することができ、ユーザが各種の指示、選択を行う際に使用される。入力部18に対して入力された指示、選択などは、電気信号に変換されてCPU16へ送られる。

## 【0033】

プログラムROM20は、携帯端末装置10の各種機能を実行するための各種

プログラムを記憶し、特に本実施形態ではビットマップフォントの変形プログラム（以下、「フォント変形プログラム」と呼ぶ。）、ビットマップフォントを利用した文字の表示プログラムなどを記憶している。

## 【0034】

フォントROM22は、ビットマップフォントの元データ（「字母データ」とも呼ぶ。）を記憶する。なお、ビットマップフォントの元データは、例えば16×16ドットなどの、縦横比が等しいフォント（「正方フォント」とも呼ぶ。）とすることが一般的である。

## 【0035】

RAM24は、ビットマップフォントの変形プログラムに従ってビットマップフォントの元データを変形処理する際に作業用メモリとして使用される。一方、変形フォントメモリ14は、ビットマップフォントの変形プログラムに従う拡大又は縮小処理により作成されたフォント（以下、「変形フォント」とも呼ぶ。）を一時的に記憶するメモリである。変形フォントメモリ14は、通常、RAMやフラッシュメモリなどにより構成することができ、携帯端末装置10が電源オフされるまで記憶内容を保持する。

## 【0036】

CPU16は、プログラムROM20内に記憶されている各種プログラムを実行することにより、携帯端末装置10の各種機能を実行する。特に、本実施形態では、プログラムROM20内に記憶されている文字表示プログラムを読み出して実行することにより、文字を表示部12上に表示させる。また、同じくプログラムROM20内に記憶されているフォント変形プログラムを読み出して実行することにより、フォントROM22内に記憶されているビットマップフォントの元データを縦方向、横方向に拡大、縮小して変形フォントを生成する。なお、CPU16は、これら以外に各種のプログラムを実行することにより携帯端末装置10の各種機能を実現するが、それらは本発明とは直接の関連を有しないので、説明を省略する。

## 【0037】

## [拡大によるフォント変形処理]

次に、本発明の特徴部分であるフォント変形処理について説明する。フォント変形処理は、前述のようにCPU16がプログラムROM20内に格納されているフォント変形小プログラムを実行することにより行われる。フォントの変形は、実際にはフォントデータを縦方向又は横方向に拡大又は縮小することにより実現される。特定のフォントの縦方向の拡大率又は縮小率と、横方向の拡大率又は縮小率が一致する場合はフォントの単純な拡大又は縮小となるが、縦方向の拡大率又は縮小率と、横方向の拡大率又は縮小率が異なる場合にフォントの変形となる。以下、フォント変形処理として、フォントの拡大による変形処理と、縮小による変形処理に分けて、順に説明する。

#### 【0038】

まず、拡大によるフォント変形処理について図2乃至図7を参照して説明する。図2乃至図4は、拡大によるフォント変形処理のメインルーチン及びサブルーチンのフローチャートである。図5乃至図7は、拡大によるフォント変形処理による各工程を説明するための図である。

#### 【0039】

図2を参照すると、まずCPU1はフォント変形指示を受け取る（ステップS1）。フォント変形指示は、例えばユーザが入力部18を操作して表示フォントの縦横比の指定、選択や表示モードの指定、選択を行った場合に、入力部18からの指示に基づいて生成される。なお、ユーザによる指示以外の場合でも、表示部12上への表示内容に応じてフォントの変形が必要となる場合もある。例えば、表示エリア内に特定の文字データを表示する際に、文字表示プログラムが自動的にフォントの変形を要求する場合もある。フォント変形指示には、縦方向の拡大率と横方向の拡大率とが含まれる。

#### 【0040】

フォントの変形指示がなされると、CPU1は表示対象となるフォントの元データをフォントROM22から読み出し、作業メモリであるRAM24へ展開する（ステップS2）。次に、CPU1は、ステップS1で入力されたフォント変形指示が、フォントの横方向の拡大を含んでいるか否かを判定する（ステップS3）。フォントの横方向の拡大を含んでいる場合（ステップS3；Yes）、CP

U1は横方向拡大処理を実行して、対象となるフォントをまず横方向に拡大する（ステップS4）。なお、ステップS3でフォント変形指示が横方向の拡大を含んでいないと判断された場合（ステップS3：No）、処理はそのまま次のステップに進む。

#### 【0041】

次に、CPU1は、ステップS1で入力されたフォント変形指示が、フォントの縦方向の拡大を含んでいるか否かを判定する（ステップS5）。フォントの縦方向の拡大を含んでいる場合（ステップS5；Yes）、CPU1は縦方向拡大処理を実行して、対象となるフォントを縦方向に拡大する（ステップS6）。なお、ステップS5でフォント変形指示が縦方向の拡大を含んでいないと判断された場合（ステップS5：No）、処理はそのまま次のステップに進む。なお、行方向の拡大率と列方向の拡大率は、それぞれステップS1で指定された割合に設定され、通常は両者は異なる。こうして、横方向及び縦方向にフォントの拡大が完了すると、CPU1は拡大後のフォントデータを変形フォントとして変形フォントメモリ14に一時的に保存し（ステップS7）、さらに表示部12上に表示する（ステップS8）。こうして、特定のフォントが変形されて表示部12上に表示される。

#### 【0042】

次に、ステップS4で行われる横方向拡大処理の詳細について図3のフローチャートを参照して説明する。図3において、まずCPU1は、対象となるフォントを複数の行に分割する（ステップS11）。そして、CPU1は、分割により得られた行ごとに、コストを計算する（ステップS12）。ここで、コストとは、フォントを構成する画素の数及び隣接する画素の有無などにより求められる値であり、拡大によるフォント変形処理においては線分量を示す。即ち、コストが大きいほど、その行には点や点の集合ではなく、線分が含まれている度合いが高いことを意味する。

#### 【0043】

図4に、ステップS12におけるコスト計算処理の詳細を示す。コスト計算処理では、まず、図5（a）に示すように、行単位でフォント構成要素の数をカウ

ントする（ステップS21）。図5（a）において、各行の下に示した数字（左から、0、4、3、3、…）が各行のコストを示している。

#### 【0044】

次に、図5（b）に示すように、隣接するフォント構成画素を有する画素分の加算を行う（ステップS22）。図5（b）の例では、隣接するフォント構成画素を有するフォント構成画素ごとに、コストに「2」を加算している。こうして、隣接画素分のコストを加算した結果得られる各行のコストが各行の下に示されている。隣接する画素を有する画素についてコストを加算することにより、その行に存在するフォント構成画素が点であるのか、線分に近いのかをある程度区分することができる。即ち、1つの行に含まれるフォント構成画素数が同じであっても、コストが大きい行ほど、その行に含まれるフォント構成画素は線分に近いということが言える。そして、後述のように、コスト値の大きい行、つまり線分に近い行ほど拡大されにくくなる。

#### 【0045】

次に、図6（a）に示すように、スペースを考慮した重み付け処理を行う（ステップS23）。具体的には、各行について、両端の画素がスペース（フォント構成画素が存在しない。図5及び6ではスペースで示されている。）である場合には、コスト値を「+2」する。また、両端の画素の1画素内側の画素がスペースである場合、コスト値を「+1」する。この処理は、文字としてのバランス及び文字を横方向に並べて文にした場合のバランスの面を考慮している。例えば平仮名の「し」という文字などは、普通に拡大すると左右の空きの部分だけが拡大されてしまい、文字を構成する中央のフォント構成画素はあまり拡大されない。その結果、拡大後の文章として見ると、他の文字は拡大されているが、文字「し」だけはフォント構成部分が十分に拡大されていないため、小さいままに見えてしまう。このような不具合を防止するために、スペースを考慮した重み付けを行うのである。この処理の結果、上下端にスペースを有する行はコスト値が大きくなり、拡大されにくくなる。

#### 【0046】

次に、同一コスト値を有する行の重み付け処理を行う（ステップS24）。具

体的には、ある行について、隣接する行とフォント構成画素のパターンを比較し、隣接する行とフォント構成画素のパターンが同じである行についてはコスト値を「+2」する。この処理も、文字としてのバランスの面を考慮して行われる。隣接する行のフォント構成画素のパターンが同一であると、これまでのコスト計算においてはコスト値は同じになる。よって、そのコスト値を有する行が後述の拡大処理により拡大対象となると、隣接する行両方が拡大されることになる。その結果、同一の画素パターンを有する行が両方とも拡大されることになり、1つの文字中の一部分のみが必要以上に拡大されてしまうことになる。そこで、隣接する行が同一のパターンを有する場合は、コスト値を増加させて、それらの行が拡大されにくすることにより、1つの文字中の特定箇所のみが偏って拡大されることを防止する。

#### 【0047】

こうして、コスト計算処理が終了すると、処理は図3に示す横方向拡大処理へ戻る。そして、CPU1は対象となるフォントの全ての行についてコスト計算が完了したか否かを判定し（ステップS13）、完了していない場合は全ての行のコスト計算を行う。全ての行についてコスト計算が完了した場合（ステップS13；Yes）、CPU1は図7(a)に示すように、コストの低い順に行を順序付けする（ステップS14）。図7(a)においては、コストの小さい行から順に、A、B、C、…と順序が付されている。なお、同じコストを有する行が複数ある場合は、文字の中央に近い行から順に上位の順序を付すことにより、拡大する方向が左右方向において偏らないようとする。図7(a)の例では、コストが「3」である列が6つあるが、それらについて中央から順に左右に分散するようにC、D、E、F、G、Hと順序付けをしている。こうして、文字の中央から左右にバランスよく拡大がなされるように、拡大対象となる順序を付けている。

#### 【0048】

コストの低い順に各行の順序付けが終了すると、CPU1は、ステップS1で指定された横方向の拡大率に基づいて、拡大する行の数を決定する（ステップS15）。この拡大率は、今回のフォント拡大処理により、ビットマップフォントの元データを横方向に拡大する割合を指す。前述のように、ステップS1におい

てフォント変形指示がなされる場合、その指示には、縦方向及び横方向の拡大率の情報が含まれている。例えば、ユーザが文字を見やすくするために、文字を縦長に変形することを指示する場合、縦方向の拡大率の方が横方向の拡大率より大きくなるし、文字を横長に変形することを指示する場合、横方向の拡大率の方が縦方向の拡大率より大きくなる。拡大率は、例えば、120%、150%などの割合として得られる。また、ユーザの指示に拘わらず、携帯端末装置10の特定の処理によって自動的にフォント変形処理がなされる場合でも、必ず縦方向及び横方向の拡大率は決定される。こうして、横方向の拡大率に基づいて、対象となるフォントの行数のうち、いくつの行を拡大するかを決定する。例えば、拡大の対象となるフォントの行数が15行（15ドット）であり、横方向の拡大率が120%（1.2倍）と決定された場合、横方向に拡大する行の数は $15 \times 1.2 = 18$ （行）となり、3つの行だけ拡大することになる。図7（a）及び（b）の例はこの場合の例である。

#### 【0049】

こうして、横方向に拡大する行の数が決定すると、CPU1は、拡大処理を実行する（ステップS16）。即ち、コストが小さい行から順に、拡大する行の数だけ拡大を行う。この場合の拡大は、例えば、拡大の対象となる行と同一のフォント画素パターンの行を拡大の対象となる行の隣に挿入することにより行われる。図7（a）及び（b）の例では、前述のように拡大する行の数は3行であるので、図7（a）に示すコストの小さい行から順に3つの行（コスト順序が「A」、「B」、「C」の行）について、同一のフォント構成画素パターンを有する行を挿入する。その結果、図7（b）に示すように、3つの行が挿入されて、全体として横方向に18行（18ドット）の拡大文字が得られている。こうして、横方向拡大処理が終了すると、処理は図2に示すメインルーチンへ戻る。

#### 【0050】

次に、図2のステップS6における縦方向拡大処理について説明する。なお、この縦方向拡大処理において対象となるフォントは、横方向への拡大処理後のフォントとなる。よって、図7（a）及び（b）の例では、横方向が18行（ドット）となったフォントに対して行われる。

## 【0051】

縦方向拡大処理は、基本的に図3に示す横方向拡大処理と同様の処理を、行毎ではなく、列ごとに行う。即ち、拡大対象となるフォントを列ごとに分割し（ステップS11）、各列についてコスト計算を行い（ステップS12）、コストの低い順に各列について順序付けを行い（ステップS14）、縦方向の拡大率に基づいて拡大対象となる列の数を決定し（ステップS15）、対象となる列に拡大処理を実行する（ステップS16）。なお、図4に示すコスト計算処理においては、縦方向に隣接画素を考慮したコストの重み付け（ステップS22）、スペースを考慮した重み付け（ステップS23）、及び同一コストを有する列の重み付け（ステップS24）を実行する。

## 【0052】

以上のように、拡大によるフォント変形処理では、対象となるフォントの構成（フォント構成画素のパターン）に基づいて行、列ごとにコストを計算し、コストに基づいて適切な行、列を拡大する。具体的には、線分に近い行、列は拡大されにくくすることにより、文字中の線分部分が必要以上に拡大されることを防止する（ステップS22）。また、端部にスペースが多い文字については、スペース部分のみが拡大されて文字部分の拡大が不十分になることを防止する（ステップS23）。さらに、同一パターンを有する行、列が隣接する場合には、それらが拡大されにくくして、文字中の特定の部分のみが必要以上に拡大されることを防止する（ステップS24）。これらの処理により、文字を構成する画素パターンを考慮して、拡大後の文字のバランスが不自然とならないようにし、自然なフォントの変形を可能としている。

## 【0053】

## [縮小によるフォント変形処理]

次に、縮小によるフォント変形処理について説明する。縮小によるフォント変形処理を図8に示す。縮小によるフォント変形処理は、基本的に拡大によるフォント変形処理と同様に行われる。即ち、CPU1は、フォント変形指示を受け取ると（ステップS41）、対象となるフォントの元データをフォントROM22から抽出し、作業メモリであるRAM24に展開する（ステップS42）。

## 【0054】

次に、CPU1は、ステップS1におけるフォント変形指示が横方向への縮小を伴うか否かを判定し（ステップS43）、横方向への縮小を伴う場合に横方向縮小処理を行う（ステップS44）。さらに、CPU1は、ステップS1におけるフォント変形指示が縦方向への縮小を伴うか否かを判定し（ステップS45）、縦方向への縮小を伴う場合に縦方向縮小処理を行う（ステップS46）。そして、変形フォントが作成されると、CPU1は、作成された変形フォントを変形フォントメモリ14に記憶する（ステップS47）とともに、表示部12に表示する（ステップS48）。こうして、縮小によるフォント変形処理がなされる。

## 【0055】

次に、縮小によるフォント変形処理における横方向縮小処理について図9を参照して説明する。図9は、横方向縮小処理のフローチャートである。図9に示す横方向縮小処理は、基本的に図3に示す拡大によるフォント変形処理における横方向拡大処理と同様の手順で行われる。即ち、CPU1は、対象となるフォントを行単位に分割し（ステップS51）、行ごとにコストを算出し（ステップS52）、全ての行についてコストが得られると（ステップS53；Yes）、コストの低い順に行を順序付けする（ステップS54）。次に、CPU1は、横方向の縮小率に基づいて縮小する行の数を決定し（ステップS55）、対象となる行について縮小処理を行う。

## 【0056】

このように、縮小によるフォント変形処理における横方向縮小処理は、基本的な流れは拡大によるフォント変形処理における横方向拡大処理と同様である。但し、縮小によるフォント変形処理における横方向縮小処理は、ステップS52におけるコスト計算方法及びステップS56における縮小方法において、拡大によるフォント変形処理の場合と異なる。以下、これについて説明する。

## 【0057】

まず、コスト計算について説明する。拡大によるフォント変形処理におけるコストは、線分量、即ち各行又は列に含まれる画素が線分に近いか、ドットに近いかを示す値であった。これに対し、縮小によるフォント変形処理におけるコスト

は、類似度、即ち各行又は列が、隣接する行又は列にどの程度類似しているかを示す値である。具体的には、CPU1は各行に対し、隣の行との排他的論理和（XOR）を算出する。排他的論理和の値は、隣接する画素が同じであると「0」となり、隣接する画素が異なると「1」となる。よって、各行に含まれる各フォント構成画素について排他的論理和を求め、その値が「1」となるフォント構成要素の数をその行のコストとする。よって、コストが高いほど、その行と隣接する行とのフォント画素パターンは非類似であり、コストが低いほど、その行と隣接する行とのフォント画素パターンは類似であることになる。こうして、コストは隣接する行との類似度（厳密には、非類似度）を示す値となる。

#### 【0058】

即ち、ある行のコストが低いということは、その行のフォント画素パターンが隣接する行のフォント画素パターンと類似していることを意味する。よって、類似している画素パターンを有する行を優先的に縮小の対象とすることにより、類似している行を削除する。これは、隣接する行が相互に類似している場合には、それを優先的に削除しても、文字全体に与える影響、違和感は少ないのであろうとの発想に基づいている。これにより、自然な縮小が可能となる。

#### 【0059】

横方向縮小処理が終わると、次に縦方向縮小処理が行われるが（ステップS46）、これは横方向縮小処理と基本的に同様である。対象となるフォントを列単位に分割し（ステップS51）、列ごとにコストを算出し（ステップS52）、全ての列についてコストが得られると（ステップS53；Yes）、コストの低い順に列を順序付けする（ステップS54）。そして、CPU1は、縦方向の縮小率に基づいて縮小する列の数を決定し（ステップS55）、対象となる列について縮小処理を行う。

#### 【0060】

図10(a)～(c)に縦方向縮小処理の様子を示す。まず、図10(a)に示すように各列についてコスト計算が行われ、次に図10(b)に示すようにコストの低い順に列の順序付けが行われ、コストが低い列から順に縮小の対象となる。ステップS56の縮小処理においては、例えば図10(c)に示すように、

コストの低い列から順にその行が削除される。図10(c)の例では、コストの低い3つの列(A、B、C)が削除されている。

## 【0061】

以上説明したように、本発明の縮小によるフォント変形処理では、対象となるフォントを行方向及び列方向に分割し、それぞれ隣接する行又は列との類似度を考慮して縮小を行う。よって、フォント画素パターンが類似する行又は列が隣接している部分を優先して縮小(行又は列の削除)を行うので、変形後のフォントが不自然となることが少なくなる。

## 【0062】

図11(a)及び(b)に、単純な座標変換によるフォント変形処理と、本発明によるフォント変形処理により得られた変形フォント(拡大例と縮小例)例を示す。なお、この例は、説明の便宜上、縦方向と横方向に同一の拡大率、縮小率で変形した例を示している。図からわかるように、単純な座標変換処理による変形フォントでは縦や横のストロークが拡大により2重になったり(「本」という文字の横線、「肌」という文字の右側の縦線など)、斜めの線がギザギザ見えたりしている。また、縮小により、文字の左右のバランスが崩れたり(「本」という文字の左右のバランス)、画素同士が連結して別の文字見えたり(「肌」という文字の左側が「月」ではなく「目」見える)している。これに対し、本発明によるフォント変形処理では、そのような不具合は生じず、自然な変形フォントが得られている。

## 【0063】

なお、以上説明したフォント変形処理は、表示部12上に表示すべき文字列が決定された時点で1文字ずつ実行することができる。また、ユーザが表示フォントの縦横比変更などを指定した場合には、フォントROM22内に予め用意されている全ての元データについてフォント変形処理を行ってその結果を変形フォントメモリ14内に格納してもよい。いずれの場合でも、上述のようにフォント変形処理自体は整数の単純な演算で済むので、処理に必要な時間は非常に短く、ユーザが不快に感じるほどの処理時間を要することはない。

## 【0064】

また、上記の説明では、フォントの変形を、拡大によるものと縮小によるものに分けて説明したが、両者を組み合わせて変形フォントを生成することも可能である。例えば、あるフォントを縦方向に拡大、横方向に縮小して縦長フォントを生成したり、縦方向に縮小、横方向に拡大して横長フォントを生成することもできる。その場合には、拡大する方向については前述の拡大によるフォント変形処理を適用し、縮小する方向には縮小によるフォント変形処理を適用すればよい。なお、図2及び図8から明らかなように、縦方向又は横方向の一方のみにフォントを拡大又は縮小して変形フォントを生成することももちろん可能であり、その場合には、拡大又は縮小する方向にのみ処理を行えばよい。

#### 【0065】

##### [横方向処理と縦方向処理の順序]

次に、フォント変形処理における横方向拡大／縮小処理と、縦方向拡大／縮小処理の順序について検討する。先に説明したフォント変形処理においては、まず横方向の拡大／縮小処理を行い、次に縦方向の拡大／縮小処理を行っている（即ち、まず横方向に拡大／縮小し、次に縦方向に拡大／縮小している）。しかし、これは1つの例であり、縦方向の拡大／縮小処理を先に行い、次に横方向の拡大／縮小処理を行うことも可能である。そして、いずれが好ましいかは、処理の対象となるフォントの構成に依存する。

#### 【0066】

図12(a)及び(b)に、同一の画素パターンに対して異なる順序で拡大を行った場合の例を示す。図12(a)の例は、まず縦方向に拡大を行い、次に横方向に拡大を行った例である。図12(a)において、最も左の元データ50aにおいて水平方向のコストを計算すると上から2行目が最も小さい。よって、上から2行目を縦方向に1画素分拡大すると真ん中のデータ50bが得られる。次に、データ50bにおいて垂直方向のコストを計算すると左から4行目及び5行目のコストが最も小さい。よって、この一方を横方向に1画素分拡大すると、最も右の拡大後データ50cが得られる。

#### 【0067】

一方、図12(b)の例は、まず横方向に拡大を行い、次に縦方向に拡大を行

った例である。図12（b）において、最も左の元データ60aにおいて垂直方向のコストを計算すると、左から4行目及び5行目が最も小さい。よって、その一方を横方向に1画素分拡大すると真ん中のデータ60bが得られる。次に、データ60bにおいて水平方向のコストを計算すると、上から1行目及び3行目のコストが最も小さい。よって、上から1行目を縦方向に1画素分拡大すると、最も右の拡大後データ60cが得られる。

#### 【0068】

このように、元データが同一でも、縦→横の順で拡大するか、横→縦の順で拡大するかによって、結果として得られるデータは異なる。図12（a）及び（b）の例では、元データに対しては図12（a）に示すように縦→横の順で拡大した方が自然に拡大がなされる。よって、全ての文字に対して自然な変形を行うためには、文字ごとに実験的に縦→横、及び横→縦の順で変形処理を行ってどちらの順序で変形処理を行う方が自然な変形フォントが得られるかを予め決めておくことが好ましい。そして、図12（c）に模式的に示すように、各フォントの元データに対して、フォント変形処理を行う場合に、縦→横の順がよいか、横→縦の順がよいかを示す処理順序情報（例えばフラグなど）を属性データとして付属させることがより好ましい。こうすれば、CPU16は、フォント変形処理の実行時にフォントROM22から読み出したフォントに付属する処理順序情報を参照し、それに示す順序でフォントの変形を行えばよい。なお、同一のフォントであっても、拡大の場合に好ましい処理順序と、縮小の場合に好ましい処理順序とが異なる場合があるので、拡大を伴う変形処理と、縮小を伴う変形処理について個別に処理順序情報を用意することが好ましい。さらに、拡大率／縮小率によって好ましい処理順序が異なる場合には、拡大率／縮小率ごとに処理順序情報を用意すればよい。

#### 【0069】

なお、本実施形態においては、コストの小さい行又は列から順に画素の拡大又は縮小処理を行うものとしたが、本発明の適用範囲はこれに限られるものではなく、例えば本実施形態において用いたコストの逆数を取ったものを新しい意味でのコストと定義すれば、コストの大きい行又は列から順に画素の拡大又は縮小処

理を行うことにより、本実施形態と同様の効果を得ることができる。

【図面の簡単な説明】

【図1】

本発明の実施形態にかかるビットマップフォントの変形処理を適用した携帯端末装置の概略構成を示す。

【図2】

本発明によるフォント変形処理のフローチャートである。

【図3】

横（縦）方向拡大処理のフローチャートである。

【図4】

拡大によるフォント変形処理におけるコスト計算処理のフローチャートである

【図5】

拡大によるフォント変形処理におけるコスト計算処理例を示す図である。

【図6】

コスト計算処理の例を示す図である。

【図7】

横（縦）方向拡大処理の例を示す図である。

【図8】

縮小によるフォント変形処理のフローチャートである。

【図9】

横（縦）方向縮小処理のフローチャートである。

【図10】

縮小によるフォント変形処理におけるコスト計算処理例を示す図である。

【図11】

本発明によるフォント変形処理と、単純な座標変換によるフォント変形処理の結果を比較する図である。

【図12】

横方向処理と縦方向処理の処理順序と処理結果との比較、及び、処理順序情報

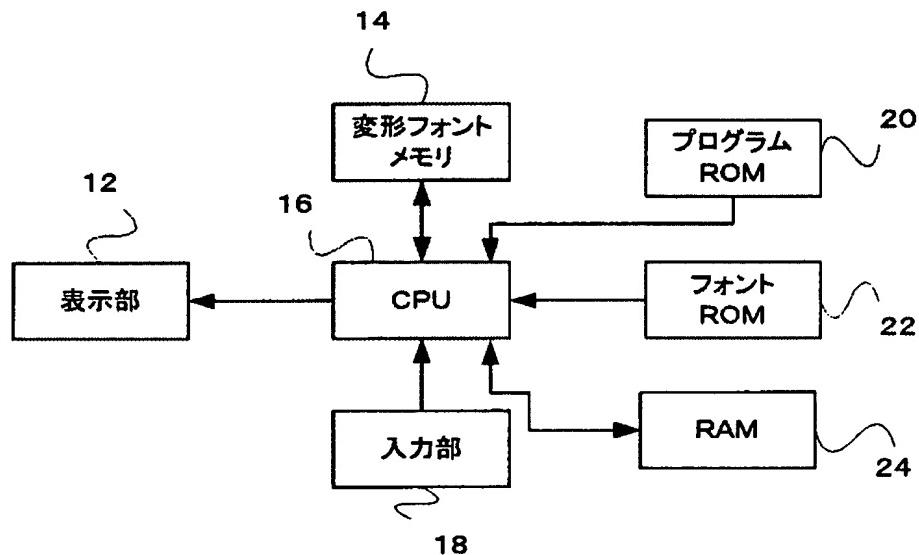
を含むフォントデータのデータ構造例を示す図である。

【符号の説明】

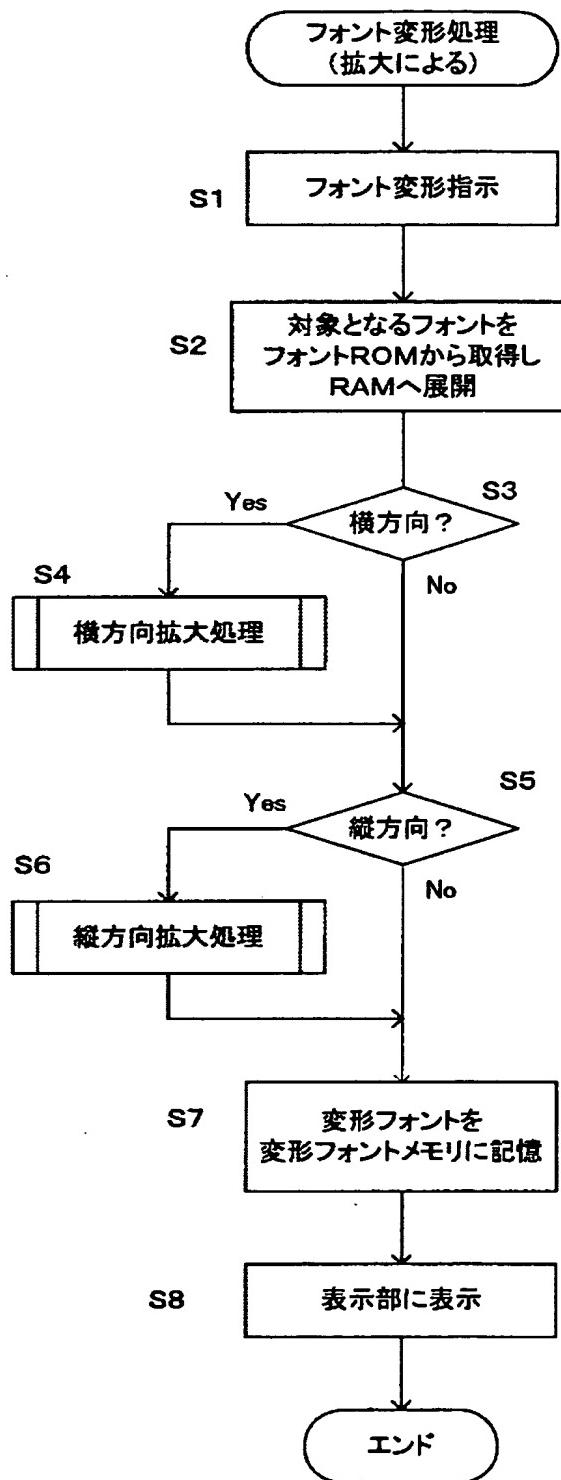
- 1 0 携帯端末装置
- 1 2 表示部
- 1 4 変形フォントメモリ
- 1 6 C P U
- 1 8 入力部
- 2 0 プログラムR O M
- 2 2 フォントR O M
- 2 4 R A M

【書類名】 図面

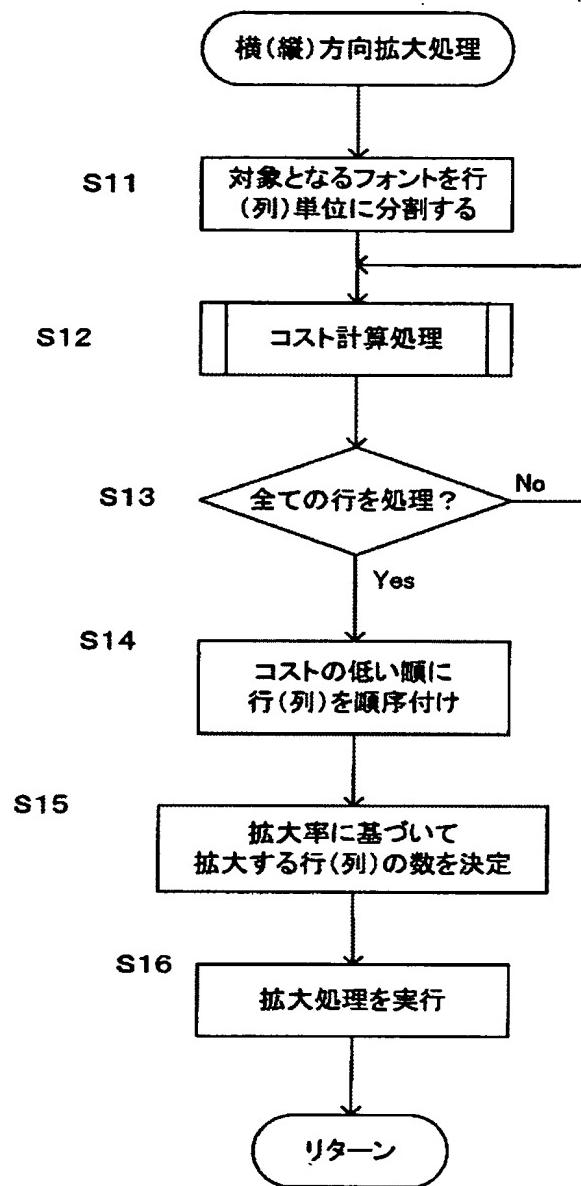
【図1】

10

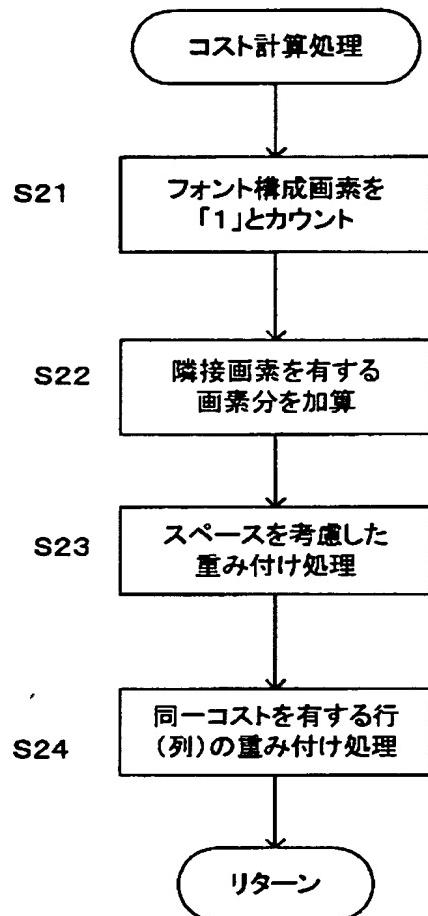
【図2】



【図3】

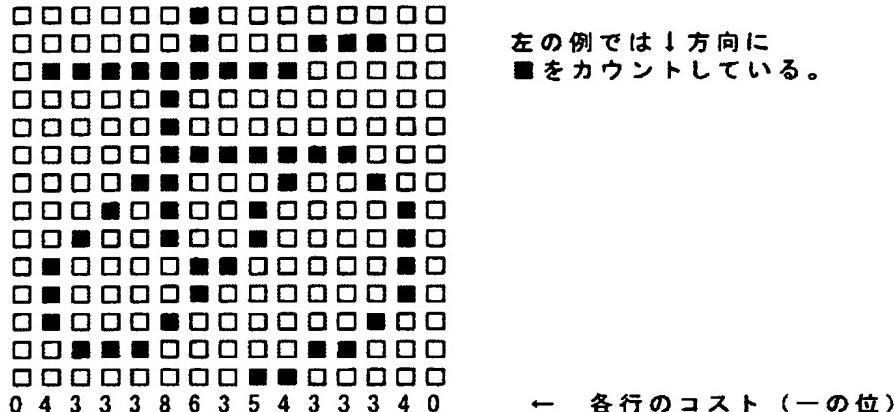


【図4】



## 【図5】

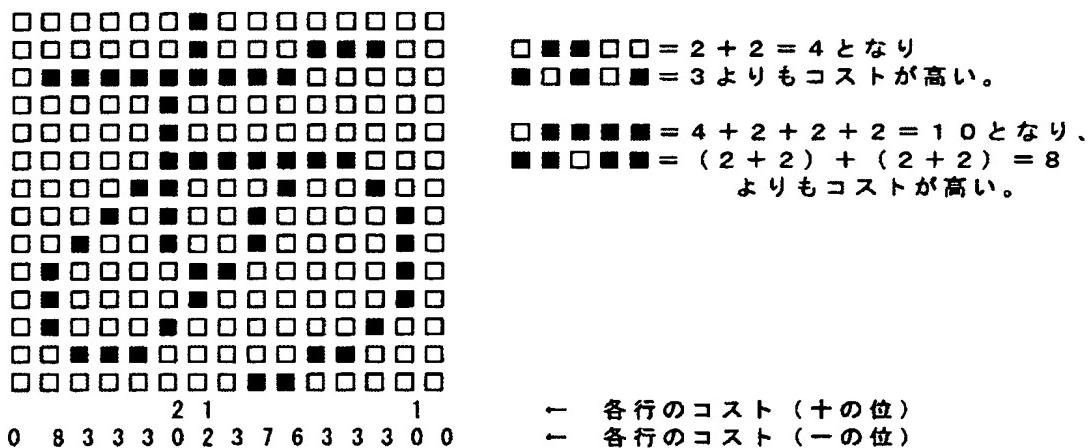
S21: フォント構成画素(■)をカウントする。



(a)

S22:隣接する画素分を加算

(連続する■は、さらに+2する)。

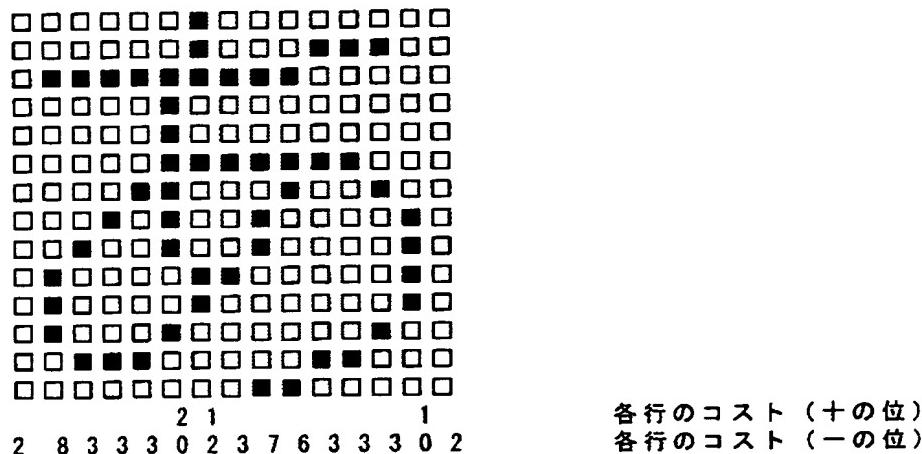


(b)

【図6】

## S23: スペースを考慮した重み付け処理

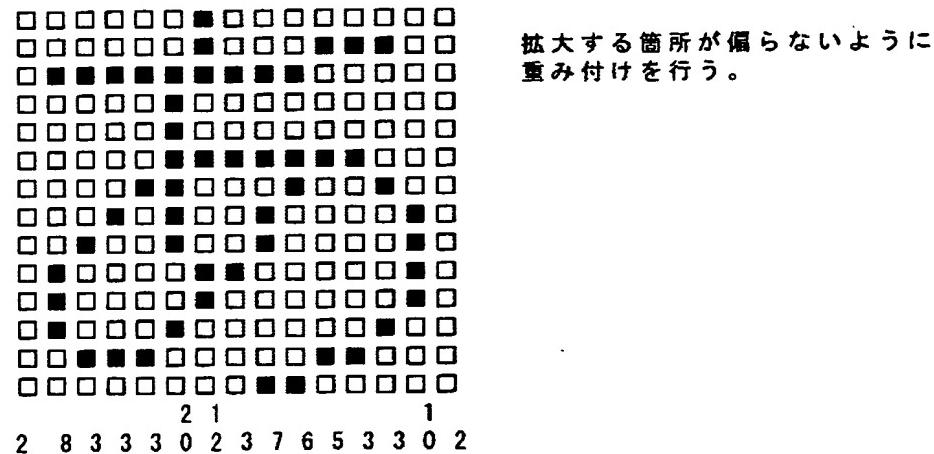
両端およびその1画素内側がスペース(■がゼロ)なら、  
それぞれ+2、+1する。



( a )

## S24: 同一コストを有する行(列)の重み付け処理

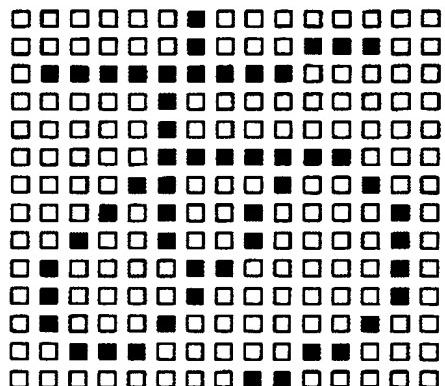
右側の列とパターンが同じなら、+2する。



( b )

## 【図7】

S14:コストが低い順に文字の中心から外側に向かって順序付け



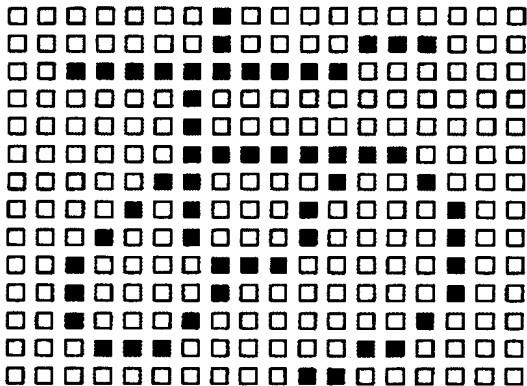
2 1                    1  
2 8 3 3 3 0 2 3 7 6 5 3 3 0 2  
B L G E D O N C K J I F H M A

拡大する方向が偏らないように、  
分散させる。

各行のコスト（十の位）  
各行のコスト（一の位）  
コストの低い順（A、B、...）

(a)

S16:拡大処理



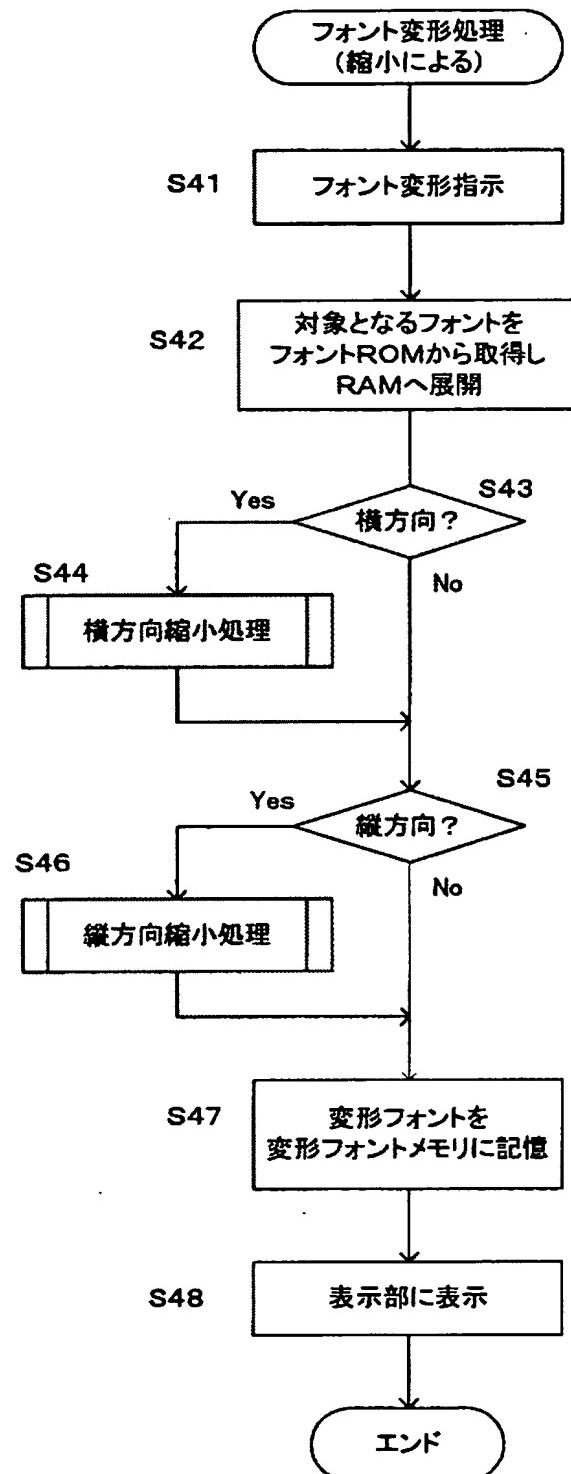
2 1                    1  
2 8 3 3 3 0 2 3 7 6 5 3 3 0 2  
B L G E D O N C K J I F H M A

左の例では、横方向に3ドット拡大  
している (15→18)。

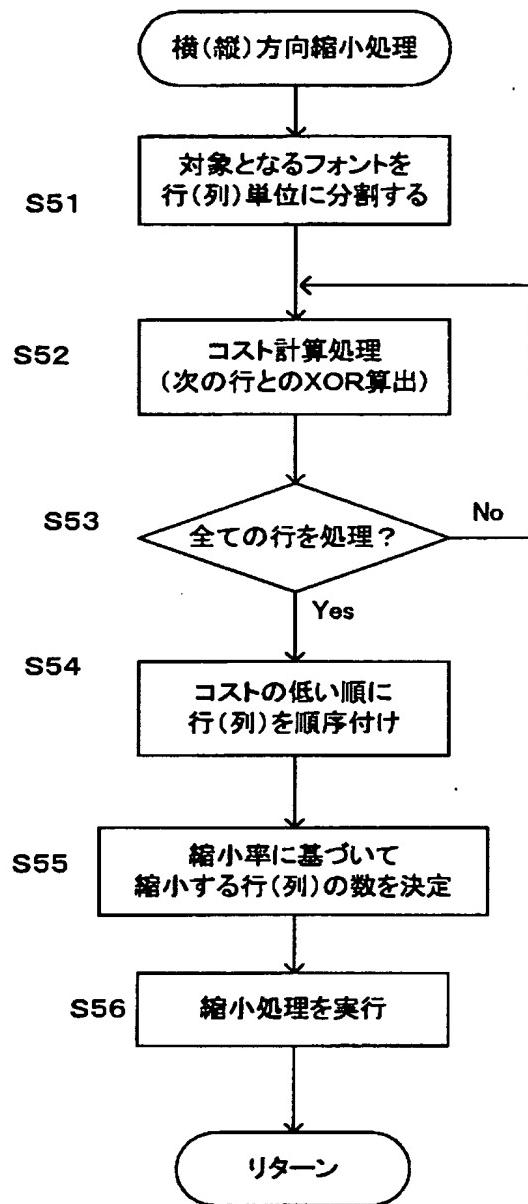
各行のコスト（十の位）  
各行のコスト（一の位）  
コストの低い順（A、B、...）

(b)

【図8】



【図9】



〔図10〕

S52:隣の列との排他的論理和(XOR)を求め、XORが1となる数を数える

左の例では下段の行との XOR を求

求め、一方向に 1 をカウント

但し、最後の行はカウントから除外

(a)

S54:コストが低い順に文字の中心から外側に向かって順序付け

A 10x10 grid of black and white squares, likely a test pattern or barcode for optical character recognition. The grid contains several black squares arranged in a non-random pattern, which are used to identify specific characters.

縮小する方向が偏らないように、分散させる。

( b )

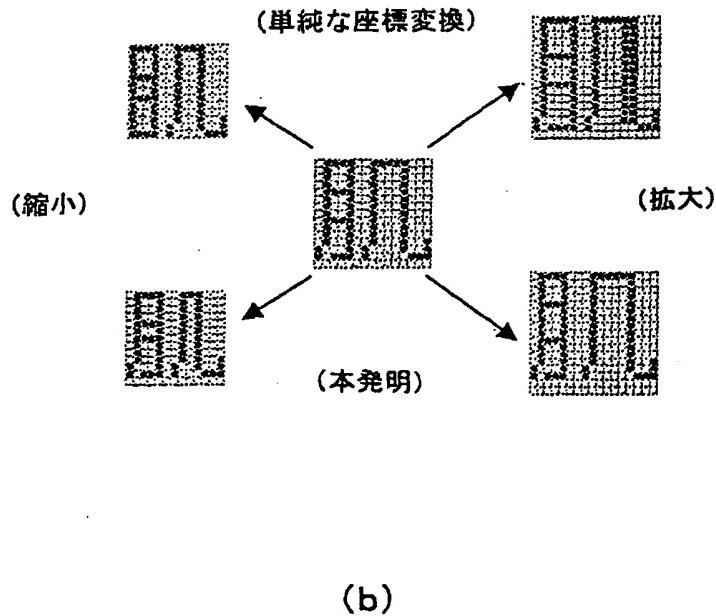
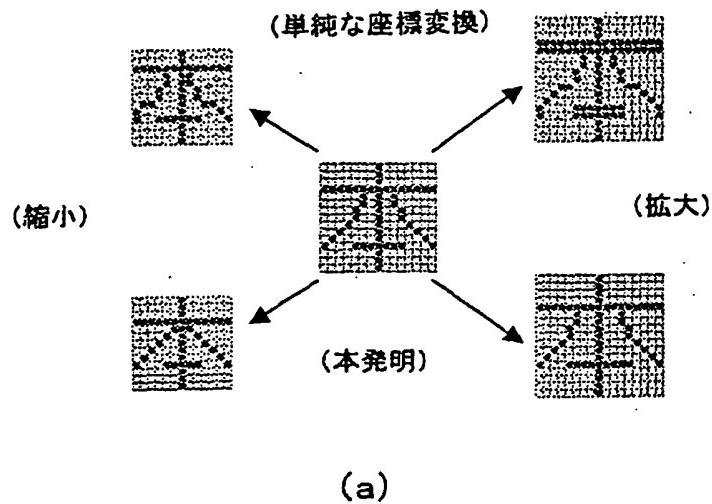
### S56：縮小處理

A 10x10 grid of black and white squares. The squares are arranged in a repeating pattern where every third square in both rows and columns is black, while the others are white. This creates a visual effect similar to a halftone dot pattern or a specific type of noise.

左の例では縦方向に3ドット縮小している(14→11)。

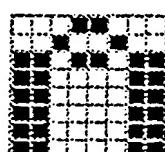
(c)

【図11】

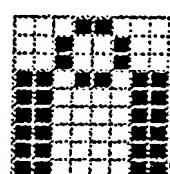


【図12】

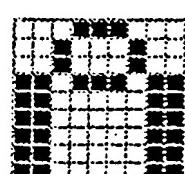
50a



50b

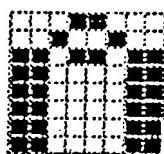


50c

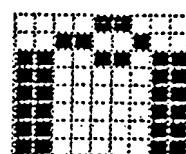


(a)

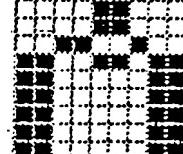
60a



60b



60c



(b)

フォントデータ  
(元データ)

処理順序情報  
(ex. フラグ)

(c)

【書類名】 要約書

【要約】

【課題】 携帯電話やPDAなどで使用するサイズの小さなピットマップフォントを自然に変形する。

【解決手段】 変形の対象となるフォントを行、列に分割し、行、列毎の構成画素パターンに基づいてコストを計算する。フォントの変形が拡大を伴う場合、拡大処理においてコストは線分量を示し、コストの小さい行又は列、即ち線分に近い画素配列を有する行又は列が拡大されにくくなる。また、フォントの変形が縮小を伴う場合、縮小処理においてコストは隣接する行、列との類似度を示し、近傍に類似する画素パターンがある行又は列は優先的に縮小の対象となる。実際に変形の対象となるフォントの画素構成に基づいて、変形の対象とする行、列を決定するので、自然な変形フォントが得られる。

【選択図】 図1

認定・付加情報

特許出願の番号	特願2002-255258
受付番号	50201300864
書類名	特許願
担当官	第一担当上席 0090
作成日	平成14年 9月 2日

<認定情報・付加情報>

【提出日】 平成14年 8月30日

次頁無

出願人履歴情報

識別番号 [000002369]

1. 変更年月日 1990年 8月20日

[変更理由] 新規登録

住 所 東京都新宿区西新宿2丁目4番1号

氏 名 セイコーエプソン株式会社